REMARKS

Reconsideration of the above-captioned application is respectfully requested. All pending claims have been rejected as being obvious in light of Messerly alone or in combination with Davis et al. Claim 7 has been amended by incorporating into it the limitations of now-canceled Claim 9, and Claim 10 has been amended to comport with the cancellation. Claims 1-8 and 10-20 remain pending.

Rejections Under 35 U.S.C. §103

Claims 1-8 and 12-18 have been rejected under 35 U.S.C. §103 as being obvious in light of Messerly, and Claims 9-11, 19, and 20 have been rejected under 35 U.S.C. §103 as being obvious in light of Messerly combined with Davis et al., used as a teaching of recovering Web pages. The limitations of the independent claims (1, 7, 12, and 15) in relation to Messerly will be discussed seriatim.

To inform the below discussion, it is important to understand what Messerly does and does not teach. In contrast to the present invention, the entire thrust of Messerly is not to manage hyperlinks, but rather simply to accumulate Web pages that are similar to each other, and then, once a broken link is found, simply return a similar page to the page pointed to by the broken link.

With this fundamental difference in mind, attention is directed to Claim 1, which requires, among other things, storing data representative of the assets and hyperlinks in a database, and then, using the database, ensuring that when a user browser selects a hyperlink represented in the database, the user is not presented with a "file not found" message. In contrast, Messerly does not teach or suggest storing hyperlinks in a database that is accessed to prevent a "file not found message", since Messerly has no need to do so.

Messerly does not contemplate managing hyperlinks. Instead, Messerly simply retrieves a similar page from the Messerly database and returns the page when a broken hyperlink is detected.

The examiner points to Messerly, col. 5, lines 44-56 and col. 6, lines 34-45 as alleged teachings of the database-stored hyperlinks. All these passages of Messerly teach, however, is what is summarized above, namely, that the Messerly database contains similar Web pages that can be identified by ID such as URL. Nowhere do the relied-upon passages mention or suggest storing hyperlinks, since there is no reason to do so under Messerly's scheme. For similar reasons, Claim 15 is not taught or suggested by Messerly. Accordingly, for at least these reasons independent claims 1 and 15 appear to be patentable over Messerly.

Applicant would also like to point out several deficiencies in the rejections of certain dependent claims. For instance, contrary to the rejection of Claim 4, Messerly cannot teach linking the data representative of the assets and hyperlinks resident in the database to the corresponding assets on the Web servers, since hyperlink data is not stored in Messerly's database, since Messerly has no need of linking its database to the servers, and since Messerly does not manage hyperlinks. Moreover, unlike Claim 5 Messerly has no need of determining that a user is attempting to create a new asset on one of the Web servers so that Messerly can crawl the new asset to identify assets and hyperlinks therein for storage thereof, for reasons set forth above.

Still further, the relied-upon passages of Messerly in rejecting Claim 6 seem far removed from the actual limitations of that claim. Specifically, nowhere in columns 4-6 does Messerly appear to teach what the examiner alleges it does, namely, determining that a user is attempting to modify an existing asset in one of the Web servers and unlinking the existing asset from the database. Neither does Messerly remotely appear to address allowing the user to update the existing asset to render a modified asset, a copy of the

existing asset being retained, crawling the modified asset to identify assets and hyperlinks therein, and storing data representative of the assets and hyperlinks of the modified asset in the database, much less relinking the modified asset and existing asset with the database. No "linking" in the context set forth in Claim 6 is taught or suggested anywhere in Messerly. For these further reasons, the dependent claims are allowable.

Turning to amended Claim 7, the examiner has failed to identify any mention in Messerly or Davis et al. (used in combination with Messerly to reject Claim 9, the limitations of which now appear in Claim 7) of linking the assets to a database containing metadata representative of the assets and reference pointers. Indeed, no mention of "metadata" in the relied-upon sections of Messerly appears at all, much less a linking that would cause backups of the database to be automatically executed so that the associated assets could be backed up on the file system or Web servers. As stated above, management of hyperlinks is simply not envisioned in Messerly. Davis et al. appears to have been used only as a general teaching of recovering Web pages, but no allegation has been made that Davis et al. teaches the claimed specific acts now recited in Claim 7, nor does it appear that Davis et al. does so. Accordingly, Claim 7 appears to be patentable.

Claim 12 requires sending metadata representative of the assets and hyperlinks to a database, whereby when a user browser selects a hyperlink ***represented in the database***, a file not found message is avoided. For the reasons set forth above, it is believed that Claim 12 is patentable over Messerly, which has no need to store any representation of hyperlinks, metadata or otherwise, in its database.

The Examiner is cordially invited to telephone the undersigned at (619) 338-8075 for any reason which would advance the instant application to allowance.

Respectfully submitted,

John L. Rogitz
Registration No. 33,549
Attorney of Record
750 B Street, Suite 3120
San Diego, CA 92101
Telephone: (619) 338-8075

JLR:jg

VERSION WITH MARKINGS TO SHOW CHANGES

7. (amended) A computer system for managing assets in a data repository such as at least one Web server or at least one file system, comprising:

computer readable code means for identifying the assets and for identifying reference pointers in the assets to other assets in the data repository;

computer readable code means for determining that a reference pointer is a broken reference pointer when the reference pointer refers to an asset not present in the data repository, such that a system manager can address the broken reference pointers, wherein the data repository includes at least one file system or at least two Web servers, and the system further comprises:

computer readable code means for linking the assets to a database containing metadata representative of the assets and reference pointers, such that backups of the database automatically cause the associated assets to be backed up on the file system or Web servers.

10. (amended) The system of Claim [9] 7, further comprising:

computer readable code means for determining that a user is attempting to create a new asset on one of the Web servers;

computer readable code means for receiving the new asset;

computer readable code means for copying the new asset to a Web server;

computer readable code means for crawling the new asset to identify assets and hyperlinks therein; and

computer readable code means for storing data representative of the assets and hyperlinks in the database.